

AMENDMENTS TO THE CLAIMS:

Please amend claims 2-20 and 41-60 as follows.

This listing of claims will replace all prior versions, and listings, of claims in the application:

1. (original) A method of selecting a performance level to be used by a data processing apparatus capable of operating at a plurality of different performance levels, said method comprising the steps of:

calculating a plurality of performance requests using respective ones of a plurality of performance request calculating algorithms;

combining said plurality of performance requests to form a global performance request;
and

selecting said performance level to be used by said data processing apparatus from among said plurality of different performance levels in dependence upon said global performance request.

2. (currently amended) A The method as claimed in claim 1, when at least one of said plurality of performance request calculating algorithms calculates a performance request independently of other of said plurality of performance request calculating algorithms.

3. (currently amended) A The method as claimed in claim 1, wherein at least one of said plurality of performance request calculating algorithms calculates a performance request based upon detected operating characteristics of said data processing apparatus.

4. (currently amended) ~~A~~The method as claimed in claim 1, wherein said step of selecting is temporally independent of said steps of calculating and combining.

5. (currently amended) ~~A~~The method as claimed in claim 1, wherein said plurality of performance request calculating algorithms are temporally independent of one another.

6. (currently amended) ~~A~~The method as claimed in claim 1, wherein said plurality of performance request calculating algorithms are associated with a hierarchy of performance request calculating algorithms, a performance request from a performance request calculating algorithm being combined with other performance requests in dependent of a position of said performance request calculating algorithm within said hierarchy of performance request calculating algorithms.

7. (currently amended) ~~A~~The method as claimed in claim 6, wherein said hierarchy of performance request calculating algorithms is fully ordered.

8. (currently amended) ~~A~~The method as claimed in claim 6, wherein said hierarchy of performance request calculating algorithms is partially ordered and an operator is provided for combining performance requests from performance request calculating algorithms on a hierarchy common level.

9. (currently amended) ~~A~~The method as claimed in claim 8, wherein said operator is a maximum value selector.

10. (currently amended) ~~A~~The method as claimed in claim 6, wherein a first priority performance request from a first priority performance request calculating algorithm can override a second priority performance request from a second priority performance request calculating algorithm when said first performance request calculating algorithm has a more dominant position within said hierarchy of performance request calculating algorithms than said second performance request calculating algorithm.

11. (currently amended) ~~A~~The method as claimed in claim 1, wherein at least one of said plurality of performance request calculating algorithms generates a command accompanying a performance request specifying how that performance request should be combined with other performance requests.

12. (currently amended) ~~A~~The method as claimed in claim 6, wherein said command specifies that said performance request should override any performance request from a performance calculating algorithm with a less dominant position in said hierarchy of performance request calculating algorithms.

13. (currently amended) ~~A~~The method as claimed in claim 6, wherein said command specifies that said performance request should be selected in preference to any lower

performance level performance request from a performance calculating algorithm in a less dominant position in said hierarchy of performance request calculating algorithms.

14. (currently amended) ~~A~~The method as claimed in claim 11, wherein said command specifies that said performance request should be ignored irrespective of any other performance level requests.

15. (currently amended) ~~A~~The method as claimed in claim 6, wherein said performance requests are combined starting from a performance request from a performance request calculating algorithm having a least dominant position within said hierarchy of performance request calculating algorithms working through to a performance request from a performance request calculating algorithm having a most dominant position within said hierarchy of performance request calculating algorithms.

16. (currently amended) ~~A~~The method as claimed in claim 1, wherein at least some of said steps calculating, combining and selecting are performed by one or more of:

an operating system kernel;
firmware of said data processing apparatus;
hardware within said data processing apparatus.

17. (currently amended) ~~A~~The method as claimed in claim 1, wherein at least one of said plurality of performance request calculating algorithms is responsive to deadline information from a real time operating system kernel.

18. (currently amended) ~~A~~The method as claimed in claim 1, wherein at least one of said plurality of performance request calculating algorithms is responsive to information from an operating system kernel.

19. (currently amended) ~~A~~The method as claimed in claim 1, wherein at least one of said plurality of performance request calculating algorithms is responsive to information from an application program, a device or a device driver.

20. (currently amended) ~~A~~The method as claimed in claim 19, wherein said information is indicative of a change in operating conditions and said at least one performance request calculating algorithm is operable to recalculate a respective one of said plurality of performance requests in response to receipt of said information.

21. (original) Apparatus for selecting a performance level to be used by a data processing apparatus capable of operating at a plurality of different performance levels, said apparatus comprising:

calculating logic operable to calculate a plurality of performance requests using respective ones of a plurality of performance request calculating algorithms;

combining logic operable to combine said plurality of performance requests to form a global performance request; and

selecting logic operable to select said performance level to be used by said data processing apparatus from among said plurality of different performance levels in dependence upon said global performance request.

22. (original) Apparatus as claimed in claim 21, when at least one of said plurality of performance request calculating algorithms calculates a performance request independently of other of said plurality of performance request calculating algorithms.

23. (original) Apparatus as claimed in claim 21, wherein at least one of said plurality of performance request calculating algorithms calculates a performance request based upon detected operating characteristics of said data processing apparatus.

24. (original) Apparatus as claimed in claim 21, wherein said step of selecting is temporally independent of said steps of calculating and combining.

25. (original) Apparatus as claimed in claim 21, wherein said plurality of performance request calculating algorithms are temporally independent of one another.

26. (original) Apparatus as claimed in claim 21, wherein said plurality of performance request calculating algorithms are associated with a hierarchy of performance request calculating algorithms, a performance request from a performance request calculating algorithm being combined with other performance requests in dependent of a position of said performance

request calculating algorithm within said hierarchy of performance request calculating algorithms.

27. (original) Apparatus as claimed in claim 26, wherein said hierarchy of performance request calculating algorithms is fully ordered.

28. (original) Apparatus as claimed in claim 26, wherein said hierarchy of performance request calculating algorithms is partially ordered and an operator is provided for combining performance requests from performance request calculating algorithms on a hierarchy common level.

29. (original) Apparatus as claimed in claim 28, wherein said operator is a maximum value selector.

30. (original) Apparatus as claimed in claim 26, wherein a first priority performance request from a first priority performance request calculating algorithm can override a second priority performance request from a second priority performance request calculating algorithm when said first performance request calculating algorithm has a more dominant position within said hierarchy of performance request calculating algorithms than said second performance request calculating algorithm.

31. (original) Apparatus as claimed in claimed claim 21, wherein at least one of said plurality of performance request calculating algorithms generates a command accompanying a

performance request specifying how that performance request should be combined with other performance requests.

32. (original) Apparatus as claimed in claim 26, wherein said command specifies that said performance request should override any performance request from a performance calculating algorithm with a less dominant position in said hierarchy of performance request calculating algorithms.

33. (original) Apparatus as claimed in claim 26, wherein said command specifies that said performance request should be selected in preference to any lower performance level performance request from a performance calculating algorithm in a less dominant position in said hierarchy of performance request calculating algorithms.

34. (original) Apparatus as claimed in claim 31, wherein said command specifies that said performance request should be ignored irrespective of any other performance level requests.

35. (original) Apparatus as claimed in claim 26, wherein said performance requests are combined starting from a performance request from a performance request calculating algorithm having a least dominant position within said hierarchy of performance request calculating algorithms working through to a performance request from a performance request calculating algorithm having a most dominant position within said hierarchy of performance request calculating algorithms.

36. (original) Apparatus as claimed in claim 21, wherein at least some of said steps calculating, combining and selecting are performed by one or more of:

- an operating system kernel;
- firmware of said data processing apparatus;
- hardware within said data processing apparatus.

37. (original) Apparatus as claimed in claim 21, wherein at least one of said plurality of performance request calculating algorithms is responsive to deadline information from a real time operating system kernel.

38. (original) Apparatus as claimed in claim 21, wherein at least one of said plurality of performance request calculating algorithms is responsive to information from an operating system kernel.

39. (original) Apparatus as claimed in claim 21, wherein at least one of said plurality of performance request calculating algorithms is responsive to information from an application program, a device or a device driver.

40. (original) Apparatus as claimed in claim 39, wherein said information is indicative of a change in operating conditions and said at least one performance request calculating algorithm is operable to recalculate a respective one of said plurality of performance requests in response to receipt of said information.

41. (currently amended) A computer program product comprising a computer readable storage medium bearing a computer program for controlling a computer to select a performance level to be used by said computer, said computer being capable of operating at a plurality of different performance levels, said computer program comprising:

calculating code operable to calculate a plurality of performance requests using respective ones of a plurality of performance request calculating algorithms;

combining code operable to combine said plurality of performance requests to form a global performance request; and

selecting code operable to select said performance level to be used by said data processing apparatus from among said plurality of different performance levels in dependence upon said global performance request.

42. (currently amended) A The computer program product as claimed in claim 41, when at least one of said plurality of performance request calculating algorithms calculates a performance request independently of other of said plurality of performance request calculating algorithms.

43. (currently amended) A The computer program product as claimed in claim 41, wherein at least one of said plurality of performance request calculating algorithms calculates a performance request based upon detected operating characteristics of said computer.

44. (currently amended) AThe computer program product as claimed in claim 41, wherein said step of selecting is temporally independent of said steps of calculating and combining.

45. (currently amended) AThe computer program product as claimed in claim 41, wherein said plurality of performance request calculating algorithms are temporally independent of one another.

46. (currently amended) AThe computer program product as claimed in claim 41, wherein said plurality of performance request calculating algorithms are associated with a hierarchy of performance request calculating algorithms, a performance request from a performance request calculating algorithm being combined with other performance requests in dependent of a position of said performance request calculating algorithm within said hierarchy of performance request calculating algorithms.

47. (currently amended) AThe computer program product as claimed in claim 46, wherein said hierarchy of performance request calculating algorithms is fully ordered.

48. (currently amended) AThe computer program product as claimed in claim 46, wherein said hierarchy of performance request calculating algorithms is partially ordered and an operator is provided for combining performance requests from performance request calculating algorithms on a hierarchy common level.

49. (currently amended) ~~A~~The computer program product as claimed in claim 48,
wherein said operator is a maximum value selector.

50. (currently amended) ~~A~~The computer program product as claimed in claim 46,
wherein a first priority performance request from a first priority performance request calculating
algorithm can override a second priority performance request from a second priority performance
request calculating algorithm when said first performance request calculating algorithm has a
more dominant position within said hierarchy of performance request calculating algorithms than
said second performance request calculating algorithm.

51. (currently amended) ~~A~~The computer program product as claimed in claim 41,
wherein at least one of said plurality of performance request calculating algorithms generates a
command accompanying a performance request specifying how that performance request should
be combined with other performance requests.

52. (currently amended) ~~A~~The computer program product as claimed in claim 46,
wherein said command specifies that said performance request should override any performance
request from a performance calculating algorithm with a less dominant position in said hierarchy
of performance request calculating algorithms.

53. (currently amended) ~~A~~The computer program product as claimed in claim 46,
wherein said command specifies that said performance request should be selected in preference

to any lower performance level performance request from a performance calculating algorithm in a less dominant position in said hierarchy of performance request calculating algorithms.

54. (currently amended) A The computer program product as claimed in claim 51, wherein said command specifies that said performance request should be ignored irrespective of any other performance level requests.

55. (currently amended) A The computer program product as claimed in claim 46, wherein said performance requests are combined starting from a performance request from a performance request calculating algorithm having a least dominant position within said hierarchy of performance request calculating algorithms working through to a performance request from a performance request calculating algorithm having a most dominant position within said hierarchy of performance request calculating algorithms.

56. (currently amended) A The computer program product as claimed in claim 46, wherein at least some of said steps calculating, combining and selecting are performed by one or more of:

- an operating system kernel;
- firmware of said data processing apparatus;
- hardware within said data processing apparatus.

57. (currently amended) AThe computer program product as claimed in claim 41,
wherein at least one of said plurality of performance request calculating algorithms is responsive
to deadline information from a real time operating system kernel.

58. (currently amended) AThe computer program product as claimed in claim 41,
wherein at least one of said plurality of performance request calculating algorithms is responsive
to information from an operating system kernel.

59. (currently amended) AThe computer program product as claimed in claim 41,
wherein at least one of said plurality of performance request calculating algorithms is responsive
to information from an application program, a device or a device driver.

60. (currently amended) AThe computer program product as claimed in claim 59,
wherein said information is indicative of a change in operating conditions and said at least one
performance request calculating algorithm is operable to recalculate a respective one of said
plurality of performance requests in response to receipt of said information.